

Titel PC-System fuer Echtzeit- und Nicht-Echtzeitprogramme						Internes Aktenzeichen 1999P03464 DE							
Prioritätsdatum		Prio-Land		Anmeldedatum 08.07.1999		Prüf-Datum		Einspruch(ja/nein) Nein		Rechtskräftige Erteilung			
Anmelder S			Bearbeiter(PIV) Wöltjen		Abteilung CT IP 1		Standort Erl S		Telefon +49 9131 7 31289		Amtl. Aktenzeichen 19931933.2		
PIK Bereichskennzeichen Wöltjen A&D MC 10						Kostenteilung 100%		Entscheidungskomplex WOEWOETNN		Fremdzahler		Offenlegungsnummer	
Zu informierende Stellen (werden auch wegen Nachanmeldung befragt)									Entwickelt im Geschäftsfeld AU211		Portfoliokomplex		
PIK Bereichskennzeichen Wöltjen A&D MC 10			Interesse Ja			Wertzahl w 4							
Erfindungsmeldung 1999E03402 DE			Figuren 4		Sig.Figur 1		Anwalt Name				Anwalt Art		

Erfinder	Dienststelle	abgegolten	abgefunden	Status	Erfinder	Dienststelle	abgegolten	abgefunden	Status
MUELLER REINER	A&DMCE39	Ja		Diensterfinder					
ULLRICH JOACHIM	RTS GMBH	Ja		Diensterfinder					
WILLERT JOSEF	A&DMCE32	Ja		Diensterfinder					

1) Kurzauszug
The PC system (PC) performs continual changeover between real-time and non real-time programs using clock-controlled switching signals. The PC system is connected to the peripheral devices (A1-An) via a communications system (B) with a cyclical communications clock that is independent of the PC system. The fixed time clock is derived from the communications clock. USE - For real-time and non real-time programs, whereby real-time programs enable control and regulation of peripherals, e.g. drives. ADVANTAGE - Enables changeover between real-time and non real-time programs in fixed time frame. Um bei einem Standard PC-System (PC), das kontinuierlich zwischen einem Echtzeitbetriebsprogramm und einem Nicht-Echtzeitbetriebsprogramm wechselt, einen für die Peripherie (A1,An) optimalen Wechsel zu erreichen, wird der Takt (T2) des Kommunikationssystems (B) zwischen dem PC-System (PC) und den peripheren Geräten (A1,An) dazu herangezogen.

Anspruch 1

Produkt Nr.	2) Produktbezeichnung
-------------	-----------------------

3) Kommentar

<p>Zugehöriges Thema: F&E-Projekt:</p> <p>im Interesse von:</p> <p>Familienübersicht Bestehende Schutzrechte: EP, EPAT, EPCH, EPDE, EPES, EPFR, EPNL, EPSE Erteilte Schutzrechte:</p> <p>Erloschene Schutzrechte: DE</p> <table style="width:100%;"> <tr> <td style="width:33%;">zugeordnete Verträge:</td> <td style="width:33%;">Vertragsart:</td> <td style="width:33%;">Vertragspartner:</td> </tr> <tr> <td>1980V01716</td> <td>GZ</td> <td>FANUC</td> </tr> <tr> <td>1997V09024/00</td> <td>EV</td> <td>RTS ECHTZEIT SOFTWARE</td> </tr> </table>	zugeordnete Verträge:	Vertragsart:	Vertragspartner:	1980V01716	GZ	FANUC	1997V09024/00	EV	RTS ECHTZEIT SOFTWARE	
zugeordnete Verträge:	Vertragsart:	Vertragspartner:								
1980V01716	GZ	FANUC								
1997V09024/00	EV	RTS ECHTZEIT SOFTWARE								

THIS PAGE BLANK (USPTO)

Beschreibung

PC-System für Echtzeit- und Nicht-Echtzeitprogramme

- 5 Die Erfindung bezieht sich auf ein Standard PC-System, bei dem über taktgesteuerte Schaltsignale kontinuierlich ein Wechsel zwischen einem Echtzeitprogramm und einem Nicht-Echtzeitprogramm innerhalb eines festen Zeittaktes erfolgt, wobei das Echtzeitprogramm die Steuerung und Regelung peripherer Geräte, insbesondere von Antrieben, erlaubt.
- 10

Für Steuerungsaufgaben werden zunehmend Standard PCs als Hardwareplattform eingesetzt. Die in diesen PCs installierten Betriebssysteme sind in der Regel für industrielle Anforderungen, beispielsweise für numerische Steuerungsaufgaben, nicht hinreichend echtzeitfähig. Die fehlende Eignung der PC-Betriebssysteme hat ihre Ursache darin, daß Dienste des Betriebssystems keine unter allen Betriebszuständen eingehaltenen maximalen Antwortzeiten garantieren.

15

Um dem abzuhelpen, kann neben dem PC-Betriebssystem für Nicht-Echtzeitprogramme ein Echtzeitbetriebssystem für Echtzeitprogramme installiert werden. Die Anwendungen, die harte Echtzeitanforderungen stellen, bedienen sich dann ausschließlich der Funktionen des Echtzeitbetriebssystems. Ein System der eingangs genannte Art hierzu ist aus der DE 196 48 422 A1 bekannt. Durch ein Taktsignal des PCs erfolgt dabei der fortlaufende Wechsel vom PC-Betriebssystem zum Echtzeitbetriebssystem.

20

Bei Echtzeitanwendungen zur Steuerung und Regelung ist ein zyklischer Datenverkehr zweckmäßig. Die Echtzeitanwendung, Ein- und Ausgabegeräte, z.B. Antriebe, bilden einen Regelkreis. Typischerweise arbeitet ein solcher Regelkreis mit einem konstanten Takt. Dieser Takt muß für alle Komponenten des Regelkreises konstant sein. Bei industrieller Anwendung ist stets eine größere Anzahl von peripheren Geräten, manchmal

25

30

35

räumlich weitläufig mit dem PC verbunden. Der Datenverkehr erfolgt dann über ein Kommunikationssystem, z.B. Profibus oder Sercos. Dieses Kommunikationssystem muß die Übertragung der Daten im Takt sicherstellen. Der Erfinder hat nun er-
5 kannt, daß für den Gesamtprozeß, d.h. auch für den Wechsel Echtzeit zu Nicht-Echtzeit, das Kommunikationssystem den Takt vorgeben sollte, da hier die kritische Prozeßführung erfolgt.

Wenn sich also die Aufgabe stellt, ein System der eingangs
10 genannten Art so auszubilden, daß ein möglichst reibungsloser Wechsel zwischen Echtzeit- und Nicht-Echtzeitanwendungen in einem industriellen Prozeß stattfinden soll, so wird diese Aufgabe gemäß der Erfindung dadurch gelöst, daß das PC-System über ein Kommunikationssystem mit einem vom PC-System unab-
15 hängigen zyklischen Kommunikationstakt mit den peripheren Geräten verbunden ist und daß aus dem Kommunikationstakt der feste Zeittakt generierbar ist.

Eine erste vorteilhafte Ausbildung der Erfindung ist dadurch
20 gekennzeichnet, daß der feste Zeittakt indirekt aus dem Kommunikationstakt generierbar sind, indem dieser zunächst hinsichtlich seiner Taktzeit egalisiert wird. Damit wird ein äquidistanter Takt für das Umschalten auch dann aufrechterhalten, wenn in der Praxis übliche Taktzeitschwankungen im
25 Kommunikationssystem auftreten.

In diesem Zusammenhang ist eine technisch äußerst einfach realisierbare Schaltungsanordnung dadurch gekennzeichnet, daß durch einen ersten Kommunikationstakt ein Zähler mit einer
30 gegenüber dem Kommunikationstakt relativ hohen Frequenz durch Rücksetzen gestartet wird, daß um den zu erwartenden Zählerstand beim Eintreffen des nächsten regulären Kommunikati- onstaktes eine untere und eine obere Komperatorschwelle gesetzt sind, daß vor Erreichen der unteren Komperatorschwelle
35 der Zähler nicht rücksetzbar ist, daß ab Erreichen der unteren Komperatorschwelle der Zähler durch einen neu eintreffenden Kommunikationstakt rücksetzbar ist, daß ansonsten bei Er-

reichen der oberen Komperatorschwelle ein automatisches Rücksetzen des Zählers erfolgt und daß durch die Rücksetzvorgänge der Zeittakt gegeben ist.

- 5 Das Umschaltsignal läßt sich aus einer solchen Schaltungsanordnung leicht dadurch erzeugen, daß bei einem definierten Zählerstand unterhalb der unteren Komperatorschwelle jeweils ein Umschaltsignal innerhalb des Zeittaktes vorgebar ist.
- 10 Ein Ausführungsbeispiel der Erfindung ist in der Zeichnung dargestellt und wird im folgenden näher erläutert. Dabei zeigen:

- FIG 1 ein Blockschaltbild,
15 FIG 2 ein erstes Funktionsdiagramm,
FIG 3 ein zweites Funktionsdiagramm und
FIG 4 ein Zählaltungsprinzip

In der Darstellung gemäß FIG 1 sind in Form eines Blockschaubildes der Übersicht halber nur die für die Erfindung wesentlichsten Elemente gezeigt. Einem Personalcomputer PC, dem ein interner Taktgeber TG1 zum Generieren eines Taktsignals T1 zur Verfügung steht, und der mit nicht geschützten Speicherbereichen S1 und geschützten Speicherbereichen S2 ausgestattet ist, werden Echtzeitprogramme EP und Nicht-Echtzeitprogramme NP zugeleitet. Über den Modus der Zuleitung wird später noch einiges berichtet, ebenso über die Berücksichtigung der Speicherbereiche S1 und S2. Die Echtzeitprogramme EP dienen zum Abarbeiten von insbesondere Aufgaben der Antriebsregelung, während die Nicht-Echtzeitprogramme NP beispielsweise Visualisierungsaufgaben erledigen können.

20
25
30

Der Wechsel zwischen Echtzeitprogramm und Nicht-Echtzeitprogramm erfolgt zyklisch, wie in FIG 2 dargestellt. Jeweils mit Beginn eines Taktes eines äquidistanten Taktsignals T, wird das Echtzeitprogramm im Personalcomputer PC aktiviert, was durch Schraffur eines Signalbalkens im oberen Teil von FIG 2

35

angedeutet ist. Beispielsweise nach vierzig Prozent der Takt-
dauer des Taktsignals T wird dann jeweils in den Echtzeitbe-
trieb gewechselt, was durch ein Umschaltsignal U ausgelöst
werden kann, wie dieses als zeitverzögertes Signal zum Takt-
5 signal T im unteren Abschnitt von FIG 2 gezeigt ist. Der je-
weils sechzigprozentige Anteil, des Nicht-Echtzeitbetriebs,
in dem der Personalcomputer PC beispielsweise im Windows NT
Betriebssystem befindlich ist, wird in der Darstellung gemäß
FIG 2 durch die freien Balkenabschnitte angedeutet.

10

Zurück zu FIG 1. Durch eine strichpunktierte Linie ist die
funktionelle Trennung des Personalcomputers PC zu einem indu-
striellen Kommunikationssystem angedeutet, bei dem ein Busma-
ster B die Kommunikationsteuerung übernimmt, damit insbeson-
15 dere den Antrieben A1 bis An im Takt des Kommunikationssy-
stems fortlaufend Steuersignale zuleitbar sind. Dazu hat der
Busmaster B einen Taktgeber TG2 für ein Taktsignal T2. Der
Busmaster kann B aber auch an einer anderen Stelle des Kommu-
nikationssystems, räumlich getrennt vom Personalcomputer PC,
20 angeordnet sein und eine Busanbindung des Personalcomputers
PC erfolgt dann über eine niederhierarchische Busanschaltbau-
gruppe, in der aber selbstverständlich der Takt T2 aus dem
Bussignal heraus detektierbar ist.

25

Die Steuerungsaufgaben und der Bustakt sind, wie eingangs
schon beschrieben wurde, für die erfindungsgemäße Arbeitswei-
se des Gesamtsystems prägend korreliert. Die Taktsignale T
und die Umschaltsignale U, wie sie in FIG 2 gezeigt sind,
werden bei der vorliegenden Erfindung nämlich aus dem Takt T2

30

des Bussystems generiert. Immer wenn ein Wechsel von einem
Nicht-Echtzeitbetrieb in einen Echtzeitbetrieb erfolgen soll
und immer wenn ein Wechsel vom Echtzeitbetrieb in den Nicht-
Echtzeitbetrieb erfolgen soll, kann dann ein Interruptsignal
über eine Interruptleitung IL ausgelöst werden und über eine

35

Interruptspezifikationsleitung IS kann mitgeteilt werden,
welcher der beiden Zustände vorliegt. Über einen Umsetzer U1
kann das Gesamtsignal dann an den Personalcomputer PC als ein

geschachtelt Interrupt gemeldet werden, denn es wird nur ein einziger Interrupt der begrenzten Anzahl der Interrupt des Personalcomputers PC benötigt und durch die Spezifikationskennung wird dennoch eine Schachtelung erreicht, die selbstverständlich für weitere Aufgaben auch tiefer gehen kann, als die im Ausführungsbeispiel benutzte Anzahl zwei.

Insbesondere bei komplexen weitläufigen Systemen muß mit signifikanten Schwankungen und eventuell sogar Unterbrechungen des Taktsignals T2 gerechnet werden. In der Darstellung gemäß FIG 3 oben ist ein derartig gestörtes Taktsignal T2 gezeigt. Damit der Personalcomputer PC diese Schwankungen jedoch nicht ausgesetzt wird, erfolgt eine Signalaufbereitung, deren Prinzip in der Schaltung gemäß FIG 4 gezeigt ist. So können die Taktsignale T2 beispielsweise durch einen Umsetzer U2 aus dem Bussignal gewonnen werden und gelangen dann über eine im Ruhezustand geschlossene Schaltstufe SS an einen Zähler Z, der in einem gegenüber dem Taktsignal T hochfrequenten Taktsignal T3 eines Taktgebers TG3 hochgezählt wird. Durch den ersten Takt des Taktsignals T2 wird der Zähler Z rückgesetzt und gestartet. Liegt an seinem Ausgang ein Zählerstand an der größer Null, aber kleiner als eine untere Komparatorschwelle K1 ist (gestrichelt in FIG 3 Mitte gezeigt), wird die Schaltstufe SS gesperrt. Ab Erreichen der unteren Komparatorschwelle K1 wird jedoch die Schaltstufe SS wieder durchgängig.

Wenn nun wieder ein Takt des Taktsignals T2 eintrifft, setzt er den Zähler zurück. Wenn jedoch dies nicht in Bälde erfolgt, erreicht der Zähler Z2 eine obere Komparatorschwelle K2 (in FIG 3 Mitte ebenfalls gestrichelt angedeutet), bei der in jeden Fall ein Rücksetzen des Zählers Z2 geschieht. Die obere und die untere Komparatorschwellen K2 und K1 sind um den Zählerstand für zeitgerecht eintreffende Taktsignale T2 herum angeordnet. Immer wenn ein Rücksetzen des Zählers Z2 erfolgt, wird ein Takt eines Taktsignals T ausgelöst.

Um auch einfach die Umschaltssignale U zu generieren, wird ein entsprechender Zählerstand für die Komperatorumschaltsschwelle K_n festgelegt, wie dies ebenfalls in FIG 3 Mitte gezeigt ist.

- 5 Alle Zählerschwellen können mit Konfigurationswerten gesetzt werden und unverändert belassen werden, es ist aber prinzipiell auch möglich, Adaptionen vorzunehmen.

10 Im Ausführungsbeispiel gemäß FIG 3 erfolgen übrigens der erste (Start) und der zweite Takt des Taktsignals T2 so, daß dieses über Rücksetzen des Zählers Z entsprechende Taktsignale T erzeugen, während der dritte Takt des Taktsignals T2 so verspätet eintrifft, daß durch zuvoriges Erreichen der oberen Komperatorschwelle K2 der dritte Takt des Taktsignals T erzeugt wird. Gleiches gilt für den vierten Takt. Der fünfte
15 Takt des Taktsignals T2 möge wieder regulär den fünften Takt des Taktsignals T auslösen.

20 Im folgenden mögen noch einige Ausführungen die Installation und Ausführung der jeweiligen Echtzeitanwendungen erläutern. Ferner wird noch einiges erläuternd zum Prinzip des geschachtelten Interrupts ausgeführt, der eingangs schon prinzipiell in seiner Wirkungsweise anhand des Ausführungsbeispiels behandelt worden ist.

25 Zum Installieren und Ausführen einer Echtzeitanwendung sei folgendes bemerkt. Das Echtzeitbetriebssystem, welches neben dem PC-Betriebssystem installiert wird, bietet ohne besondere Vorkehrungen keine Möglichkeit, interaktiv Echtzeitanwendungen zu starten, zu beenden oder in anderer Form zu beeinflussen.
30 Die nachfolgend beschriebene Technik erlaubt das Starten einer Echtzeitanwendung in einem wie oben beschriebenen PC-System mit Echtzeiterweiterung durch Mechanismen des PC-Betriebssystems.

35 Die Echtzeitanwendung wird mit einem Entwicklungssystem für PC Anwendungen erstellt. Diese Anwendung kann mit den Stan-

dardmechanismen des PC-Betriebssystems gestartet werden. Für das PC-Betriebssystem handelt es sich bei der Echtzeitanwendung um ein gewöhnliches Programm. Nach dem Start der Echtzeitanwendung durchläuft diese eine Initialisierungsphase, in welcher noch keine echtzeitrelevanten Aktivitäten ausgeführt werden. In diesem Programmabschnitt meldet sich die Echtzeitanwendung über die Schnittstelle des Echtzeitbetriebssystems bei diesem an. Bei der Anmeldung werden die Eintrittspunkte in die echtzeitrelevanten Programmteile mitgeteilt. Nach Durchlaufen dieses Programmteils ist die Echtzeitanwendung bereit für den durch externe Signale gesteuerten Betrieb.

Wie gesagt, sind geschützte und ungeschützte Speicherbereiche S2 und S1 beim Personalcomputer PC vorgesehen. Zur Ausführung der Echtzeitanwendung in geschützter Umgebung sei daher noch folgendes bemerkt.

In PC Betriebssystemen (insbesondere Windows NT) wird jedes Programm in einer Umgebung ausgeführt, die unerlaubte Zugriffe zwischen Programmen und unerlaubte Zugriffe auf Systemdaten und Betriebsmittel verhindert (user-mode). Eine Echtzeitanwendung, die sich auf einen Echtzeitbetriebssystem abstützt, muß dagegen uneingeschränkten Zugriff auf alle vom Echtzeitsystem verwalteten Betriebsmittel haben. Das bedeutet, daß die Echtzeitanwendung in ungeschützter Umgebung (kernelmode) ablaufen muß. Ein Wechsel einer Anwendung vom geschützten in den ungeschützten Mode wird von den PC Betriebssystemen (Windows NT) nicht unterstützt.

Gemäß der Erfindung kann die folgend beschriebene Technik benutzt werden, damit unter dem PC Betriebssystem Windows NT ein Wechsel der Echtzeitanwendung vom geschützten in den ungeschützten Mode erreicht wird.

Für die Echtzeiterweiterung des PC Betriebssystems wird ein Gerätetreiber realisiert. Ein Gerätetreiber stellt eine Erweiterung des Betriebssystems dar. Das Programm des Geräte-

treibers wird immer im ungeschützten (kernel-) Mode ausgeführt. Dieser Gerätetreiber der Echtzeiterweiterung bietet eine Schnittstelle, über die eine Anwendung (insbesondere auch eine Echtzeitanwendung) dem Gerätetreiber Funktionen (der Anwendung) mitteilen kann, die dieser dann aus dem ungeschützten Mode heraus aktivieren kann. (Zur Erklärung: Wird aus dem ungeschützten Mode eine Funktion aufgerufen, die zu einer Anwendung und damit zu einem geschützten Programmteil gehört, bleibt der ungeschützte Mode weiter aktiv.)

10

Die Ausführung von Funktionen der Echtzeitanwendung im ungeschützten Mode kann synchron erfolgen, d.h. die von der Echtzeitanwendung an den Treiber mitgeteilte Funktion wird direkt aktiviert, während der rufende Programmteil der Echtzeitanwendung an der Programmstelle stehen bleibt bis, das Programm aus der Treiberfunktion zurückkehrt. Eine solche synchrone Ausführung kann eingesetzt werden, wenn nach der Anmeldung der Echtzeitanwendung die weitere Initialisierung im ungeschützten Mode erfolgen soll (um z.B. uneingeschränkt auf die Betriebsmittel der Echtzeitanwendung zugreifen zu können).

20

Eine asynchrone Ausführung im ungeschützten Mode wird erreicht, wenn im Treiber die Funktionen der Echtzeitanwendung gespeichert werden und auf ein Signal hin diese Funktionen ausgeführt werden. Das Umschalten vom PC Betriebssystem in das Echtzeitbetriebssystem ist ein solcher asynchroner Aufruf einer Funktion des Echtzeitbetriebssystems im ungeschützten Mode: Das Umschaltsignal führt zur Aktivierung seiner zugeordneten Interruptbehandlungsroutine. Diese ist Teil des Treiberprogramms und wird im ungeschützten Mode ausgeführt. In dieser Interruptbehandlungsroutine erfolgt ein Wechsel in den Kontext der Echtzeitanwendung und eine dem Umschaltsignal zugeordnete Funktion der Echtzeitanwendung wird aktiviert.

30

Zu den geschachtelten Interrupts kann noch folgendes ergänzend ausgeführt werden. In einem PC System steht nur eine begrenzte Anzahl von Signalen für die Interruptbehandlung frei

35

zur Verfügung. Von den fünfzehn Interruptsignalen des Standard-PCs sind die meisten durch Standardgeräte des PC-Systems belegt. Eine Echtzeitanwendung benötigt in der Regel mehrere Signale, auf die es mit kurzer Latenzzeit reagieren muß.

5

Übliche Praxis ist es, daß alle Signale, auf die die Echtzeitanwendung reagieren muß, durch eine Hardwareschaltung in einem Register gespeichert werden und jede Änderung im Register als ein Sammelsignal an den Interruptmechanismus des PCs weitergeleitet werden. In der diesem Sammelsignal zugeordneten Interruptbehandlungsroutine wird das Register ausgelesen und dadurch die Quelle der Interruptanforderung ermittelt. Bei einer größeren Anzahl von Echtzeitsignalen ist es notwendig, diese priorisiert zu behandeln, wobei höher priorisierte Echtzeitsignale die Behandlung von Echtzeitsignalen niedrigerer Priorität verdrängen.

Mehrere Echtzeitsignale über einen einzigen Interrupteingang des Standard PC zu behandeln, macht es erforderlich, daß diese geschachtelt abgearbeitet werden können. Die Interruptbehandlung von Windows NT sieht dieses nicht vor. Eine geschachtelte Behandlung von Interruptanforderungen auf der gleichen Interruptleitung wird durch erfindungsgemäße Technik auch unter PC Betriebssystemen möglich, die dieses nicht vorsehen.

Die Interruptbehandlungsroutine, die dem Sammelsignal der Echtzeitsignale zugeordnet ist, wird direkt in die Interrupt-Tabelle des Prozessors eingetragen. Dadurch wird die primäre Behandlung der Interruptanforderung durch Windows NT umgangen und muß durch eine entsprechende Behandlung in der Interruptbehandlungsroutine ersetzt werden. Insbesondere quittiert die Interruptbehandlungsroutine die Interruptanforderung beim Interruptmechanismus des PC, wodurch diese bereit gemacht wird, auf der gleichen Interruptleitung eine weitere Anforderung zu erzeugen. Die Interruptbehandlungsroutine, muß wiedereintrittssicher implementiert sein.

Patentansprüche

1. Standard PC-System, bei dem über taktgesteuerte Schaltsi-
gnale kontinuierlich ein Wechsel zwischen einem Echtzeitpro-
gramm und einem Nicht-Echtzeitprogramm innerhalb eines festen
Zeittaktes erfolgt, wobei das Echtzeitprogramm die Steuerung
und Regelung peripherer Geräte, insbesondere von Antrieben,
erlaubt, dadurch gekennzeichnet, daß das PC-System (PC,S1,S2)
über ein Kommunikationssystem (B) mit einem vom PC-System
(PC,S1,S2) unabhängigen zyklischen Kommunikationstakt (T2)
mit den peripheren Geräten (A1,An) verbunden ist und daß aus
dem Kommunikationstakt (T2) der feste Zeittakt (T) generier-
bar ist.
2. System nach Anspruch 1, d a d u r c h g e k e n n -
z e i c h n e t , daß der feste Zeittakt (T) indirekt aus
dem Kommunikationstakt (T2) generierbar sind, indem dieser
zunächst hinsichtlich seiner Taktzeit egalisiert wird.
3. System nach Anspruch 2, d a d u r c h g e k e n n -
z e i c h n e t , daß durch einen ersten Kommunikationstakt
ein Zähler (Z) mit einer gegenüber dem Kommunikationstakt
(T2) relativ hohen Frequenz durch Rücksetzen gestartet wird,
daß um den zu erwartenden Zählerstand beim Eintreffen des
nächsten regulären Kommunikationstaktes eine untere und eine
obere Komperatorschwelle (K1,K2) gesetzt sind, daß vor Errei-
chen der unteren Komperatorschwelle (K1) der Zähler (Z) nicht
rücksetzbar ist, daß ab Erreichen der unteren Komperator-
schwelle (K1) der Zähler (Z) durch einen neu eintreffenden
Kommunikationstakt rücksetzbar ist, daß ansonsten bei Errei-
chen der oberen Komperatorschwelle (K2) ein automatisches
Rücksetzen des Zählers (Z) erfolgt und daß durch die Rück-
setzvorgänge der Zeittakt (T) gegeben ist.

4. System nach Anspruch 3, d a d u r c h g e k e n n -
z e i c h n e t , daß bei einem definierbaren Zählerstand
(KU) unterhalb der unteren Komperatorschwelle (Kl) jeweils
ein Umschaltsignal (U) innerhalb des Zeittaktes (T) vorgebbar
5 ist.

Zusammenfassung

PC-System für Echtzeit- und Nicht-Echtzeitprogramme

- 5 Um bei einem Standard PC-System (PC), das kontinuierlich zwischen einem Echtzeitbetriebsprogramm und einem Nicht-Echtzeitbetriebsprogramm wechselt, einen für die Peripherie (A1,An) optimalen Wechsel zu erreichen, wird der Takt (T2) des Kommunikationssystems (B) zwischen dem PC-System (PC) und
- 10 den peripheren Geräten (A1,An) dazu herangezogen.

FIG 1

1/2

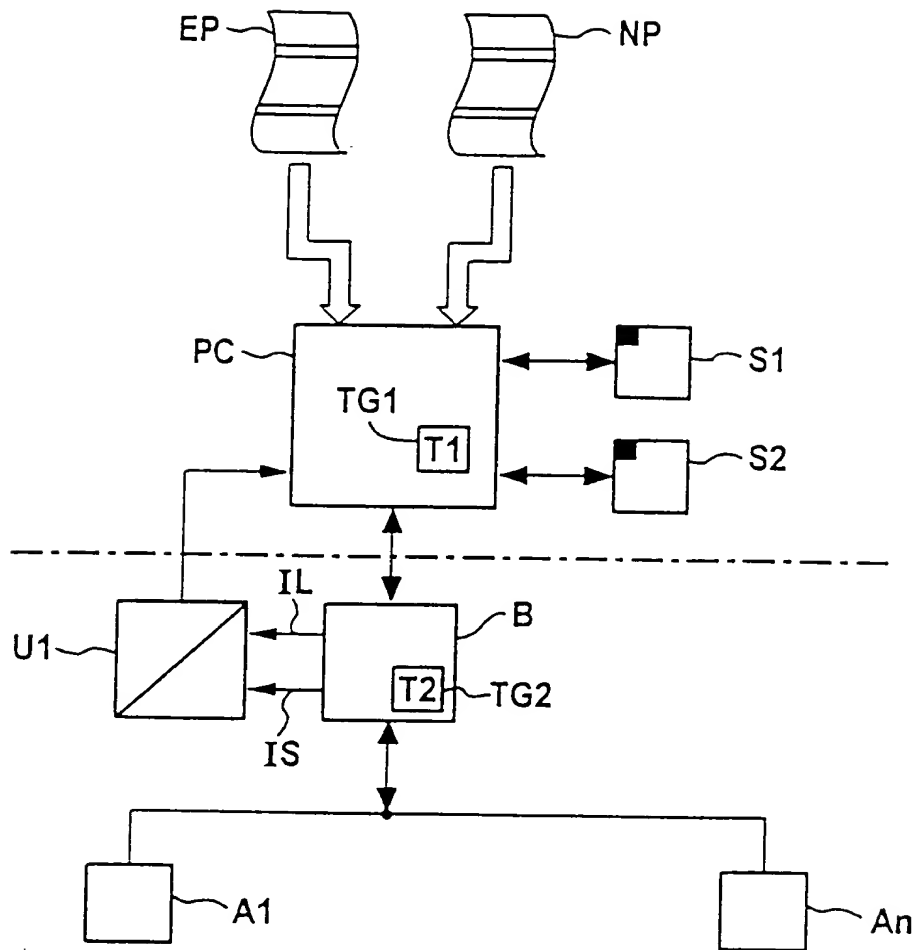


FIG 1

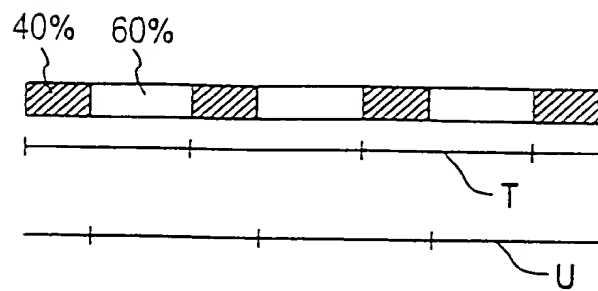


FIG 2

2/2

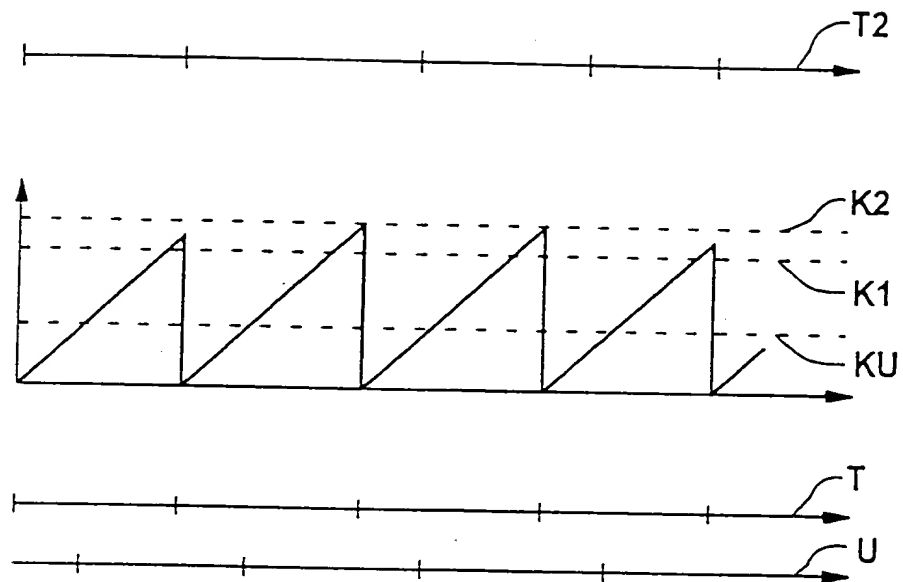


FIG 3

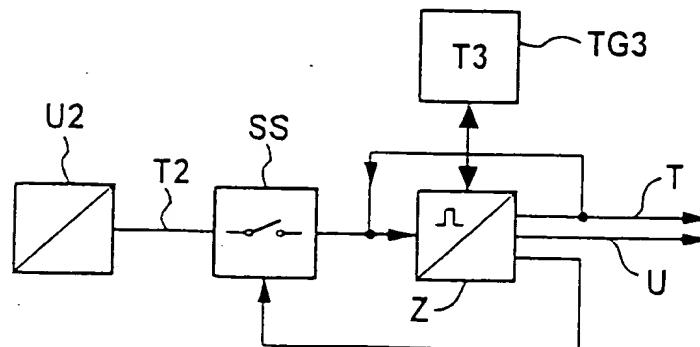


FIG 4